

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-294069

(43)Date of publication of application : 11.11.1997

(51)Int.Cl. H03K 19/173
 H03K 19/173
 G06F 15/18
 H01L 21/82
 // G06F 7/00

(21)Application number : 09-027797

(71)Applicant : AGENCY OF IND SCIENCE & TECHNOL

(22)Date of filing : 12.02.1997

(72)Inventor : HIGUCHI TETSUYA
 MURAKAWA MASAHIRO

(30)Priority

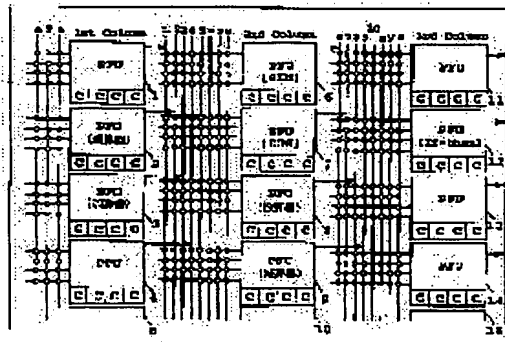
Priority number : 08 45223 Priority date : 01.03.1996 Priority country : JP

(54) PROGRAMMABLE LSI AND ITS ARITHMETIC METHOD

(57)Abstract:

PROBLEM TO BE SOLVED: To program a numerical arithmetic expression by designating the type of an arithmetic circuit included in an arithmetic unit based on a desired arithmetic expression and connecting the arithmetic circuit via a connector means.

SOLUTION: Each of function units PFU 1 to 15 has plural arithmetic circuits of different arithmetic contents. An arithmetical operation circuit, etc., can be constituted by a memory which contains about 10 types of function tables for addition, subtraction, multiplication, subtraction, SIN subtraction, COS subtraction, etc. In such a constitution, the arithmetic result is outputted to the data line of the memory from a designated area when the data designating a desired function table and the data on the arithmetic object are inputted to the address line of the memory. Furthermore, the logical arithmetic tables of AND, OR, etc., are prepared together with an arithmetic circuit which carries out a logical operation of an IF-THEN form. Then the executing arithmetic circuit is selected for every memory and computing element by a switch of a multiplexer, etc., and based on the digital signal that is externally pointed.

**LEGAL STATUS**

[Date of request for examination] 12.02.1997

[Date of sending the examiner's decision of rejection] 27.07.1999

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3170599

[Date of registration] 23.03.2001

[Number of appeal against examiner's decision of rejection] 11-13505

[Date of requesting appeal against examiner's decision of rejection] 25.08.1999

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

*** NOTICES ***

JPO and NCIP I are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] Programmable LSI characterized by having two or more arithmetic units which each has the arithmetic circuit plurality from which a class differs, receive from the outside assignment of the arithmetic circuit with which activation of an operation is presented, input the data used for an operation, and output the result of an operation, and a connecting means for connecting mutually according to the operation expression which defined said two or more arithmetic units beforehand.

[Claim 2] It is programmable LSI characterized by said connecting means being a crossbar switch in programmable LSI according to claim 1.

[Claim 3] Programmable LSI characterized by including the circuit which performs math processing in said two or more arithmetic circuits in programmable LSI according to claim 1.

[Claim 4] Programmable LSI characterized by including the circuit which performs logical operation in said two or more arithmetic circuits in programmable LSI according to claim 1.

[Claim 5] Corresponding to the contents of the operation which the instruction which has the memory which memorizes the instruction which directs the contents of the operation to which said arithmetic unit is given from the outside in programmable LSI according to claim 1, and was memorized by this memory directs, it is programmable LSI characterized by choosing said two or more arithmetic circuits.

[Claim 6] It is programmable LSI characterized by calculating by the arithmetic circuit which has two or more storage regions for said memory to memorize said two or more instructions in programmable LSI according to claim 5, read said instruction from two or more storage regions concerned one by one, and was chosen according to the read instruction concerned.

[Claim 7] It is programmable LSI characterized by said connecting means enabling connection of said two or more arithmetic units with a matrix gestalt in programmable LSI according to claim 1.

[Claim 8] Programmable LSI characterized by enabling connection of said two or more arithmetic units by the tree structure in programmable LSI according to claim 1.

[Claim 9] When said connecting means carries out an adjustable setup of the number of hierarchies of said tree structure in programmable LSI according to claim 8, it is programmable LSI characterized by connecting said two or more arithmetic units alternatively.

[Claim 10] It is programmable LSI characterized by carrying out an adjustable setup of the number of hierarchies of said tree structure by locating said connecting means on each hierarchy of said tree structure in programmable LSI according to claim 9, having the selection circuitry which chooses the result of an operation from the specific arithmetic unit which adjoins on the hierarchy of a top or the bottom, and choosing the result of an operation by this selection circuitry.

[Claim 11] Two or more arithmetic units which each has the arithmetic circuit plurality from which a class differs, receive from the outside assignment of the arithmetic circuit with which activation of an operation is presented, input the data used for an operation, and output the result of an operation. The connection circuit for connecting mutually according to the operation expression which defined said two or more arithmetic units beforehand is LSI-ized. The operation approach of programmable LSI characterized by directing the arithmetic unit connected with the arithmetic circuit performed with said two or more arithmetic units, and said crossbar switch according to the operation expression defined beforehand.

[Claim 12] The operation approach of programmable LSI characterized by changing the contents of said operation expression one by one by changing the initial data in which the class of arithmetic circuit given to each aforementioned arithmetic unit in the operation approach of programmable LSI according to claim 11 is shown one by one.

[Claim 13] The operation approach of programmable LSI characterized by determining the contents of suitable operation expression using the technique of a genetic algorithm in the operation approach of programmable LSI according to claim 11.

[Claim 14] The arithmetic unit of the downstream which each of said arithmetic unit sets the flag information which shows operation termination during an operation in the operation approach of programmable LSI according to claim 11 according to an operation condition, and is connected to this arithmetic unit is the operation approach of programmable LSI characterized by incorporating the result of an operation of the upstream of connection based on said flag.

[Claim 15] Corresponding to the contents of the operation which the instruction which has the memory which

memorizes the instruction with which each of two or more of said arithmetic circuit units directs the contents of the operation in the operation approach of programmable LSI according to claim 11, and was memorized by this memory directs, it is the operation approach of programmable LSI characterized by choosing said two or more arithmetic circuits.

[Claim 16] It is the operation approach of programmable LSI which said memory has two or more storage regions for memorizing said two or more instructions, reads said instruction from two or more storage regions concerned one by one in the operation approach of programmable LSI according to claim 11, and is characterized by calculating with the selected computing element.

[Claim 17] It is the operation approach of programmable LSI characterized by said connection circuit enabling connection of said two or more arithmetic units with a matrix gestalt in the operation approach of programmable LSI according to claim 11.

[Claim 18] The operation approach of programmable LSI characterized by enabling connection of said two or more arithmetic units by the tree structure in the operation approach of programmable LSI according to claim 11.

[Claim 19] When said connection circuit carries out an adjustable setup of the number of hierarchies of said tree structure in the operation approach of programmable LSI according to claim 18, it is the operation approach of programmable LSI characterized by connecting said two or more arithmetic circuits alternatively.

[Claim 20] It is the operation approach of programmable LSI characterized by carrying out an adjustable setup of the number of hierarchies of said tree structure by locating said connecting means on each hierarchy of said tree structure in programmable LSI according to claim 19, having the selection circuitry which chooses the result of an operation from the specific arithmetic circuit which adjoins on the hierarchy of a top or the bottom, and choosing the result of an operation by this selection circuitry.

[Translation done.]

*** NOTICES ***

JPO and NCIPJ are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.**** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]**[0001]**

[Field of the Invention] This invention relates operation expression to programmable (programmable) programmable LSI (large-scale integrated circuit) and its operation approach.

[0002]

[Description of the Prior Art] As this kind LSI, FPGA (Field ProgrammableGate Array) is known well conventionally. FPGA consists of crossbar switches which connect alternatively two or more programmable Logical unit and these programmable units. Logical unit -- and (AND) -- or logical operation, such as (OR), is performed.

[0003] In addition, LSI which combined the adder subtracter according to CPU (Central Processing Unit), DSP (Digital Signal Processor), or operation expression as a circuit which performs four operations etc., and carried out non-switched connection in dedication is known.

[0004]

[Problem(s) to be Solved by the Invention] Logical operation is often performed in the above-mentioned conventional example, and, as for FPGA, math processing is difficult. On the other hand, although math processing is possible for CPU or DSP, since the operation expression which specified math processing is prescribed by the program, in order to change operation expression, the time and effort of rewriting the whole program is needed. Moreover, LSI which combined computing elements, such as addition and subtraction, cannot perform modification of operation expression, namely, has the fault of not being programmable.

[0005] Then, the purpose of this invention is to offer suitable programmable LSI to program a math-processing type and its operation approach in view of an above-mentioned fault.

[0006]

[Means for Solving the Problem] It is characterized by to have a connecting means for connecting mutually according to the operation expression which defined beforehand two or more arithmetic units which each has the arithmetic circuit plurality from which a class differs, invention of claim 1 receives from the outside assignment of the arithmetic circuit with which activation of an operation is presented, inputs the data used for an operation, and output the result of an operation, and said two or more arithmetic units in order to attain such a purpose.

[0007] Invention of claim 2 is characterized by said connecting means being a crossbar switch in programmable LSI according to claim 1.

[0008] Invention of claim 3 is characterized by including the circuit which performs math processing in said two or more arithmetic circuits in programmable LSI according to claim 1.

[0009] Invention of claim 4 is characterized by including the circuit which performs logical operation in said two or more arithmetic circuits in programmable LSI according to claim 1.

[0010] Invention of claim 5 is characterized by said arithmetic unit choosing said two or more arithmetic circuits corresponding to the contents of the operation which the instruction which has the memory which memorizes the instruction which directs the contents of the operation given from the outside, and was memorized by this memory directs in programmable LSI according to claim 1.

[0011] In programmable LSI according to claim 5, said memory has two or more storage regions for memorizing said two or more instructions, and reads said instruction from two or more storage regions concerned one by one, and invention of claim 6 is characterized by calculating by the arithmetic circuit chosen according to the read instruction concerned.

[0012] Invention of claim 7 is characterized by said connecting means enabling connection of said two or more arithmetic units with a matrix gestalt in programmable LSI according to claim 1.

[0013] Invention of claim 8 is characterized by enabling connection of said two or more arithmetic units by the tree structure in programmable LSI according to claim 1.

[0014] As for said connecting means, invention of claim 9 is characterized by connecting said two or more arithmetic units alternatively in programmable LSI according to claim 8 by carrying out an adjustable setup of the number of hierarchies of said tree structure.

[0015] Invention of claim 10 is characterized by said connecting means carrying out an adjustable setup of the number of hierarchies of said tree structure by being located on each hierarchy of said tree structure, having the selection circuitry which chooses the result of an operation from the specific arithmetic unit which adjoins on the hierarchy of a top or the bottom, and choosing the result of an operation by this selection circuitry in

programmable LSI according to claim 9.

[0016] Two or more arithmetic units which each has the arithmetic circuit plurality from which a class differs, and invention of claim 11 receives from the outside assignment of the arithmetic circuit with which activation of an operation is presented, input the data used for an operation, and output the result of an operation. The connection circuit for connecting mutually according to the operation expression which defined said two or more arithmetic units beforehand is LSI-ized, and it is characterized by directing the arithmetic unit connected with the arithmetic circuit performed with said two or more arithmetic units, and said crossbar switch according to the operation expression defined beforehand.

[0017] Invention of claim 12 is characterized by acquiring the solution of the operation expression defined beforehand in the operation approach of programmable LSI according to claim 11 by changing the initial data given to said arithmetic unit of the maximum upstream one by one.

[0018] Invention of claim 13 is characterized by using the technique of a genetic algorithm, in order to acquire said solution.

[0019] Invention of claim 14 sets the flag information each of said arithmetic unit indicates operation termination during an operation to be in the operation approach of programmable LSI according to claim 11 according to an operation condition, and the arithmetic unit of the downstream linked to this arithmetic unit is characterized by reading the result of an operation of the upstream of connection based on said flag.

[0020] Invention of claim 15 is characterized by each of two or more of said operation time units choosing said two or more arithmetic circuits corresponding to the contents of the operation which the instruction which has the memory which memorizes the instruction which directs the contents of the operation, and was memorized by this memory directs in the operation approach of programmable LSI according to claim 11.

[0021] Invention of claim 16 is characterized by for said memory having two or more storage regions for memorizing said two or more instructions, reading said instruction from two or more storage regions concerned one by one, and calculating with the selected computing element in the operation approach of programmable LSI according to claim 11.

[0022] Invention of claim 17 is characterized by said connection circuit enabling connection of said two or more arithmetic units with a matrix gestalt in the operation approach of programmable LSI according to claim 11.

[0023] Invention of claim 18 is characterized by enabling connection of said two or more arithmetic units by the tree structure in the operation approach of programmable LSI according to claim 11.

[0024] As for said connection circuit, invention of claim 19 is characterized by connecting said two or more arithmetic circuits alternatively in the operation approach of programmable LSI according to claim 18 by carrying out an adjustable setup of the number of hierarchies of said tree structure.

[0025] Invention of claim 20 is characterized by said connecting means carrying out an adjustable setup of the number of hierarchies of said tree structure by being located on each hierarchy of said tree structure, having the selection circuitry which chooses the result of an operation from the specific arithmetic circuit which adjoins on the hierarchy of a top or the bottom, and choosing the result of an operation by this selection circuitry in the operation approach of programmable LSI according to claim 19.

[0026]

[Embodiment of the Invention] Hereafter, the gestalt of operation of this invention is explained to a detail with reference to a drawing. Drawing 1 shows the basic configuration of the circuit which the gestalt of this operation LSI-izes. In drawing 1, 1-15 are function units (PFU). Each function unit (arithmetic unit of this invention) can use the same thing. Each of a function unit has two or more arithmetic circuits where the contents of an operation differ. Four-operations ***** etc. can consist of memory. In memory, about ten kinds of function tables, such as the object for addition, an object for subtraction, an object for multiplication, an object for subtraction, an object for the operation of SIN, and an object for the operation of COS, are stored.

[0027] The numeric value 6 of an addition result is stored in the memory area where the function table for addition becomes settled by numeric values 5 and 1 in the case of 5+1. Therefore, if the data which specify a function table to use, and the data for an operation (in the case of the above-mentioned addition numeric values 5 and 1) are inputted into the address line of memory, the result of an operation will be outputted to the data line of memory from the memory area specified in the address. Each above-mentioned PFU also stores the constant table besides an above-mentioned operation table, and the output of only a constant is possible for it. Moreover, logical operation tables and IF, such as AND and OR It also has the arithmetic circuit which performs logical operation of a THEN format. The arithmetic circuit with which activation is presented by switchers, such as a multiplexer, is chosen by the digital signal by which such memory and a computing element are directed from the outside. in addition, the thing for which the arithmetic circuit of other gestalten may be used although the arithmetic circuit which uses memory with the gestalt of this operation was made into the example — it is natural.

[0028] Each PFU interconnects alternatively with a crossbar switch (O mark in drawing illustrates). In addition, it is shown that connect with input data x at y, and PFU2 which the part shown by — expresses connection, for example, is located in eye the 1st train (1stColumn) inputs these data. It is shown that PFU6 of eye the 2nd train (2ndColumn) connects with PFU2 of eye the 1st train, and inputs the operation output of PFU2.

[0029] Therefore, a user can specify the class of operation which each PFU is made to perform using non-illustrated a DIP switch and other designating devices, and the operation expression of arbitration can be set up by operating a crossbar switch. The operation expression incidentally set up in the example of connection of

drawing 1 is [0030].

[Equation 1]

IF $\cos(X+Y) > \sin(X*Z')$ then $Z'-Y$ else $(X+Y)/Y$ is shown. It means that this formula makes the value of $Z'-Y$ the result of an operation when the value of $\cos(X+Y)$ is larger than the value of $\sin(X*Z')$, and it makes the value of $(X+Y)/Y$ the result of an operation when that is not right.

[0031] The example which showed such operation expression in the combination of the conventional computing element is shown in drawing 2. In the former, if the circuit shown in drawing 2 is LSI-ized, although it is not easy to change *****, be careful of the point which can rearrange operation expression freely with the gestalt of this operation.

[0032] Thus, the time amount which performs programmed operation expression becomes as follows. They are [count / sin] 1 unit time and if to 5 unit time and subtraction in 5 unit time and cos count. Supposing it makes it then count in 2 unit time and makes 3 unit time important point to 2 unit time and a division at multiplication, the time amount which the timing of the operation performed by each PFU and the whole operation take comes to be shown in drawing 3. Since the partial operation which constitutes operation expression from a gestalt of this operation can be performed in juxtaposition as shown in drawing 3, the gestalt of this operation has processing speed quicker than the arithmetic unit which performs the partial operation in serial like CPU.

[0033] Next, the concrete example of a configuration of programmable LSI is explained using drawing 9.

[0034] In drawing 9, the PFU block 100 is a chip which has 15-set Mino PFU. The PFU controller 200 and a system controller 300 are prepared by the PFU block 100 and the pair. A system controller 300 inputs a chip select signal (it is written as CS signal), a lead signal (it is written as RD signal) or a light signal (it is written as WR signal), and a system address signal from the exterior, and supplies RD signal or WR signal to the PFU block 100 chosen by CS signal. RD signal is a signal which specifies the address of the PFU instruction memory within a PFU block, or the below-mentioned register in PFU, and is System at the time of generating of RD signal. Data are read from the storage region (plurality) and the above-mentioned register of the instruction memory (it installs in each PFU) as which the address was specified by the addr signal, and it is System. It is outputted to a Data signal line. The data read are usually data handed over for other chips (PFU block).

[0035] Conversely, at the time of generating of WD signal, it is System. An executive program instruction and operation data (operand) are written in to the storage region and the above-mentioned register of instruction memory as which the address was specified by the Addr signal. The instruction memory 116 reaches the executive program instruction about each PFU, and memorizes data. There are three sorts of instructions, operation instruction, control instruction, and immediate instruction, among the executive program instructions. The function for operation instruction to perform eight sorts of operations, such as four operations, with the gestalt of this operation is defined.

[0036] Control instruction performs directions of I/O of the operation data and the result of an operation to a specific register, assignment of the decimal point location, assignment of branch condition, etc. Immediate instruction is data which should be calculated.

[0037] IN0-IN7 are 8 sets of input data which should be processed with the PFU block 100.

[0038] The PFU controller 100 is Sytem. The information transmitted by the data signal is analyzed, the below-mentioned signal is created, and it supplies to PFU within the PFU block 100. This supply signal is explained.

[0039] It is the signal which shows which function of the eight functions in which the instruction execution of each PFU is possible a func signal chooses, and a func signal is sent to each PFU. For example, the PFU changes the function which carries out instruction execution to multiplication from addition by rewriting the contents of the func signal over left-hand side PFU in the condition that three PFU as shown in drawing 16 (e) is connected. A modification result becomes drawing 16 (f).

[0040] PFU go signal is a signal which directs initiation of operation to all PFU.

[0041] PFU A sel signal is supplied to each PFU by the signal by which actuation is permitted, and PFU to which this signal is not set cannot operate.

[0042] A mux signal is a signal which directs the connection configuration of two or more PFU within the PFU block 100. It is controlled by this signal the signal line of two or more PFU un-connecting [connection /], for example, the connection structure of various kinds of trees as shown in (a) of drawing 15, (b), (c), and (d) is built.

[0043] PFU A done signal is an activation terminate signal sent from each PFU.

[0044] CLK An enable signal is a timing signal of operation created from CLK (reference clock signal), and PFU operates synchronizing with this signal.

[0045] The main configurations about PFU within the PFU block 100 are shown in drawing 10. In drawing 10, it carries out as each result of an operation of PFU0 and PFU1 is inputted into PFU8 and each result of an operation of PFU2 and PFU3 is inputted into PFU9, and by the gestalt of drawing 10, PFU is connected on the hierarchy (tree) of four layers. Furthermore in drawing 16, the result of an operation of specific PFU (PFU0, PFU8, PFU12, PFU14) of each left end hierarchy's contiguity can be taken out. A desired thing is taken out by the multiplexer (called MPX, a brief sketch, and a selector) 101 out of this taken-out result of an operation. Please note the point which does not have the switch which performs connection/** in the signal line between two PFU with the gestalt of this operation.

[0046] It is directed by the mux signal above-mentioned [of which PFU MPX101 takes out the result of an operation]. For example, what is necessary is to choose the result of an operation of PFU14 in MPX101 and just to output to calculate by the tree structure of drawing 15 as shown in (a). The result of an operation of

PFU12 is chosen to calculate by the tree structure of drawing 15 as shown in (b). In the case of a tree structure like drawing 15 (c), the result of an operation of PFU8 is chosen, and, in the case of the tree structure of drawing 15 (d), the result of an operation of PFU0 is chosen.

[0047] By considering as such a connection configuration, a switch group becomes unnecessary and circuitry is simplified. Moreover, also when constructing operation expression predetermined by two or more PFU, there is an advantage that it is easy to grasp correspondence relation with the function assigned to each PFU and its PFU.

[0048] An example of the configuration of above-mentioned PFU is shown in drawing 11, and an internal configuration is shown.

[0049] PFU mainly consists of FIFO buffer 110, the register groups 111, the multipliers (MPU) 114, the logic units (ALU) 115, and the PFU instruction memory 116 for data inputs. Above-mentioned System addr It is System by addressing by the signal. The contents of the data signal are written in from the outside. The instruction stored in the instruction memory 116 by addressing from an address control circuit (Next Address Control) through the register PC for program counters is read one by one, and the contents of the instruction are analyzed by the decoder 117. MPU114, ALU115, and the register group 111 are controlled by this analysis result, and the operation by the selected function is performed.

[0050] An address stack is a store circuit for memorizing the return address, when a subroutine (branch instruction) is given, and the return address of the address which a stack pointer (SP) directs is read from an address stack. Since the operation of eight sorts of functions is possible for PFU as mentioned above, the both sides of MPY114, ALU115, or MPY114 and ALU115 are chosen by the class of selected function.

[0051] In relation to this operation, the data used for an operation from FIFO110 for external inputs and the predetermined register in the register group 111 are transmitted to MPY114 and ALU115 through bus A-BUS, B-BUS, and C-BUS. Moreover, it is also possible to input the result of an operation of MPY114 into ALU115. Selector groups 112 and 113 are used for the data input from an above-mentioned register and FIFO.

[0052] It is possible for MPY114 to have Inputs R and S in this example, and for the result of an operation to be stored in Register M, and to store the result of an operation of MPY14 in registers M0-M2 in the path of Mout->M0-M2. It is possible for ALU115 to have Inputs U and V, and for the result of an operation to be stored in Register A, and to store the result of an operation of ALU15 in registers A0-A2 in the path of register Aout->A0-A2. When outputting these results of an operation, a shifter is used for adjustment of the decimal point location. Register M, Register A, and MPY14 and ALU15 are both reset through Registers Mrs and Ars by the external reset signal or the signal from an address control circuit.

[0053] Registers X and Y store the data of an external input, and registers c0-c8 store a constant. With the gestalt of this operation, the chromosome data explained with the front operation gestalt are stored. An immediate is stored in Register im.

[0054] These configuration sections are CLK. Actuation becomes possible with enable signals created from the enable signal, such as a signal (for example, EN0, ENX, etc.) which has the notation of EN.

[0055] I/O-related explanation of an above-mentioned circuit of operation is explained with reference to drawing 12 R> 2, drawing 13, and drawing 14. Drawing 12 shows the signal generation timing when carrying out an information input from the outside. Drawing 13 shows the signal generation timing in the case of outputting the result of an operation. Drawing 14 shows the signal generation timing in the case of performing processing initiation and a halt of PFU.

[0056] In drawing 12, when the instruction read from the instruction memory 116 in time of day T1 is an input instruction (the store instruction of Register IR inputs), information is inputted into FIFO by the side of Register X by the ENX signal, and information is inputted into FIFO by the side of Register Y by the ENY signal by time-of-day T3. According to generating of generating of the EmptyX signal in time of day T2, the EmptyY signal in time-of-day T four, and an ENI signal, it is transmitted to Registers X and Y from two FIFO110 at input.

[0057] On the other hand, it is CLK when an output instruction is read from the instruction memory 116 (the store instruction of Register IR outputs), as shown in drawing 13. EN0 signal occurs synchronizing with an enable signal, and the result of an operation is outputted from Register Mout or Register Aout.

[0058] It is PFU as it indicates drawing 14 that initiation of operation was suited. According to generating of go signal, ink MENTO of the value of the register PC for program counters is carried out one by one, and program instruction is read from the instruction memory 116 one by one. PFU It responds to elimination of go signal and is FIFO. A reset signal is generated and FIFO110 is reset. Moreover, other circuits suspend actuation.

[0059] Detailed explanation will not be required since the timing of operation relevant to an operation is the same as that of the conventional arithmetic circuit.

[0060] As mentioned above, as explained, with an above-mentioned operation gestalt, two or more PFU is connected by the tree structure, and compared with the connection structure of PFU of a matrix gestalt, a connection configuration can be simplified more by enabling an adjustable setup of the hierarchy.

[0061] In addition, in the example shown in drawing 10, without limiting to this example, although the example of the PFU block which constructs PFU 15 and has it was shown, a request can construct PFU and it can be made into a number. The following example other than the gestalt of above-mentioned operation is realizable.

[0062] 1) although it is the example which acquires the result of an operation of operation expression by giving the input data which programs operation expression and is substituted for the gestalt of this operation at operation expression, by repeating and giving the group of input data and the data of the result of an operation, it comes out to make it change by study, and it can do operation expression.

[0063] Thus, in case it asks for suitable operation expression, the technique of a genetic algorithm (Genetic Algorithm) can be used. Although this algorithm is already proposed by the invention-in-this-application person and is announced ("hardware evolving" 1995BIT (the October issue)), it introduces these contents of a proposal simply.

[0064] Connection relation between the class of arithmetic circuit where it was chosen in each arithmetic unit, and an arithmetic unit is made into a solution, and the candidate of this solution is expressed as a chromosome (0, binary bit string of 1), two or more these is prepared, and it considers as an initial ensemble. Moreover, the candidate of these solutions is called an individual. For example, one of the black dots on the curve shown in drawing 4 is expressed as a binary bit string of 0 and 1 which is looked at by drawing 5, and two or more (drawing 5 four pieces) these is prepared.

[0065] The performance index which defines the goodness of a solution is defined and let the value be fitness. The optimal performance index is prepared for every problem to solve.

[0066] The high individuals of fitness are chosen from ensembles. An each object is chosen according to the rate that an each object occupies general one as the approach of the selection in total of the fitness in an ensemble by the approach by proportional distribution. A rate is shown in drawing 5 and drawing 6.

[0067] It applies to two individuals which made [above-mentioned] selection of a crossover (what moved by turns as some data showed drawing 7), or the mutation (that from which some data changed to other things), and a still better individual is made. Thus, screening the low solution of whenever [application / which becomes settled by the performance index in the made solution], and keeping the number of ensembles of a solution constant, it repeats selection of the above-mentioned individual, and a structure substitute of an individual until the solution which whenever [application] is high and is satisfied is acquired.

[0068] Consequently, the operation expression set up in programmable LSI is changed automatically, and the suitable operation expression for last is obtained. In addition to pattern recognition, as an application of this programmable LSI, it can be used for control of the data compression in decryption, the statistical analysis of data, and a communication link, and ATM (exchange).

[0069] 2) Usually, operations, such as addition and multiplication, respond seeds and the operation times differ. So, in each PFU of drawing 1, it is good to prepare the flag information which shows an operation condition and to tell PFU of the downstream about an operation condition. PFU of the upstream sets up the flag which expresses operation termination with a bit 1 and more specifically expresses under an operation with a bit 0 at the time of operation initiation and termination. This flag information is set as a flip-flop or a latch circuit. In PFU of the downstream, this flag information is supervised with procedure as shown in drawing 8 (loop-formation processing of drawing 8 of step S1 → S2), and when flag information shows count termination, that result of an operation is inputted from PFU of the upstream (step S3). Moreover, the flag information of self is set as 0 from self for PFU of the downstream. If the operation of the directed class is performed and the result of an operation is obtained, a count result will be set to the buffer for an output for handing over the result of an operation to PFU of the downstream, and it will set to the flag information 1 of self (step S4 → S5). When each PFU performs such processing, it becomes it is asynchronous and possible to transmit data.

[0070]

[Effect of the Invention] As mentioned above, as explained, class assignment is carried out according to the operation expression of a request of the arithmetic circuit in an arithmetic unit, and by connecting by the connecting means (crossbar switch), a math-processing type can be programmed and it can go by invention of claims 1, 2, 3, and 11.

[0071] In invention of claim 4, it becomes possible to program the logical operation type using a math-processing result by including a logical circuit into an arithmetic circuit.

[0072] In invention of claims 5 and 15, by having memory, each arithmetic unit can receive different ***** from the outside, and can fix and change the contents of an operation to perform.

[0073] Since an operation which memorizes collectively the instruction which directs contents of an operation which are different in invention of claims 6 and 16 when memory has two or more storage regions in memory, and is different one by one can be performed, other type and various operation expression can be performed and operation expression becomes programmable.

[0074] By invention of claims 7 and 17, all operations can be performed by making the topology of an arithmetic unit into a matrix gestalt.

[0075] At claims 8-10 and invention of 18-20, an operation can be performed with arithmetic units fewer than a matrix gestalt by making the topology of an arithmetic unit into a tree structure. Moreover, it can respond to different operation expression by carrying out adjustable [of the number of hierarchies of a tree structure]. Moreover, by choosing the result of an operation of each hierarchy's specific arithmetic circuit, it is not necessary to form especially the switch that performs connection/** of a signal line, the circuitry inside LSI can be simplified, and it can have, and can contribute to a system-wide miniaturization.

[0076] The solution to, i.e., suitable operation expression, is acquirable with invention of claim 12 by changing the initial data which specifies the class of each operation of operation expression, and its connection one by one. A neural network and a statistical analysis circuit can be constituted from programmable LSI by this, and pattern recognition, data analysis, etc. can be performed.

[0077] In invention of claim 13, a suitable solution can be chosen in the acquired solution.

[0078] In invention of claim 14, it is asynchronous and data transfer between arithmetic units can be performed.

[Translation done.]

* NOTICES *

JPO and NCIPJ are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is the block diagram showing the circuitry of the gestalt of this invention operation.

[Drawing 2] It is the block diagram showing the example of connection of the arithmetic circuit according to operation expression.

[Drawing 3] It is the explanatory view showing data-processing timing.

[Drawing 4] It is an explanatory view for explaining a genetic algorithm.

[Drawing 5] It is an explanatory view for explaining a genetic algorithm.

[Drawing 6] It is an explanatory view for explaining a genetic algorithm.

[Drawing 7] It is an explanatory view for explaining a genetic algorithm.

[Drawing 8] It is the flow chart which shows the contents of data communication processing between PFU.

[Drawing 9] It is the block diagram showing the system configuration of the gestalt of other operations.

[Drawing 10] It is the block diagram showing the connection configuration of PFU.

[Drawing 11] It is the block diagram showing the internal configuration of PFU.

[Drawing 12] It is the timing chart which shows the signal generation timing of an input.

[Drawing 13] It is the timing chart which shows the signal generation timing of an output.

[Drawing 14] It is the timing chart which shows the signal generation timing of initiation of operation and a halt.

[Drawing 15] It is the explanatory view showing the example of a connection pattern of PFU.

[Drawing 16] It is the explanatory view showing a switch of the contents of an operation of PFU.

[Description of Notations]

1-15 Function unit

100 PFU Block

110 FIFO (Buffer)

111 Register Group

112,113 Selector

114 Multiplier (MPU)

115 Logic Unit (ALU)

116 Instruction Memory

117 Decoder

200 PFU Controller

300 System Controller

[Translation done.]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-294069

(43) 公開日 平成9年(1997)11月11日

(51) Int.Cl. ⁸	識別記号	庁内整理番号	F I	技術表示箇所
H 0 3 K 19/173	1 0 1		H 0 3 K 19/173	1 0 1
G 0 6 F 15/18	5 5 0		G 0 6 F 15/18	5 5 0 C
H 0 1 L 21/82			H 0 1 L 21/82	A
// G 0 6 F 7/00			G 0 6 F 7/00	E

審査請求 有 請求項の数20 O L (全 12 頁)

(21) 出願番号 特願平9-27797

(22) 出願日 平成9年(1997)2月12日

(31) 優先権主張番号 特願平8-45223

(32) 優先日 平8(1996)3月1日

(33) 優先権主張国 日本 (J P)

(71) 出願人 000001144

工業技術院長

東京都千代田区霞が関1丁目3番1号

(72) 発明者 樋口 哲也

茨城県つくば市梅園1丁目1番4 工業技術院電子技術総合研究所内

(72) 発明者 村川 正宏

埼玉県南埼玉郡白岡町新白岡3-4-5

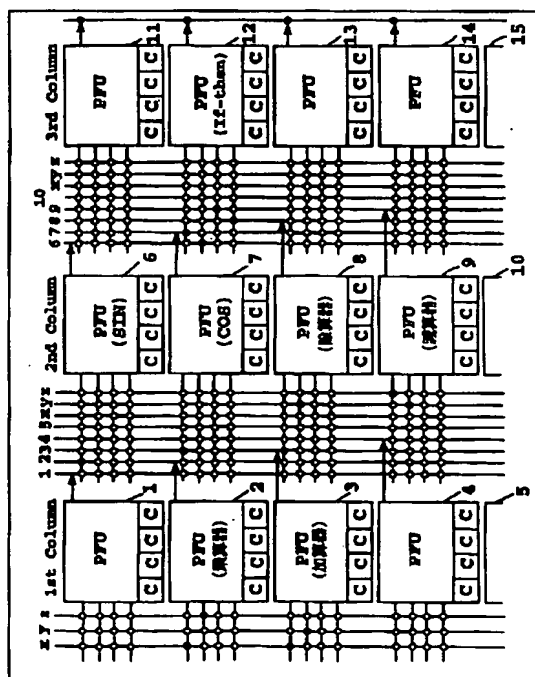
(74) 指定代理人 工業技術院電子技術総合研究所長

(54) 【発明の名称】 プログラマブルLSIおよびその演算方法

(57) 【要約】

【課題】 数値演算式をプログラマブルなLSIを提供する。

【解決手段】 異なる演算回路複数を切り換え的に選択する関数ユニット1~15を用意し、各関数ユニットに対して演算式に対応する四則演算等の種類を指定する。また、演算式の演算の順序に従って関数ユニットをクロスバスイッチにより接続する。



【特許請求の範囲】

【請求項1】 種類の異なる演算回路複数をそれぞれが有し、演算の実行に供する演算回路の指定を外部から受け付け、演算に使用するデータを入力し、演算結果を出力する複数の演算ユニットと、前記複数の演算ユニットを予め定めた演算式に従って相互に接続するための接続手段とを具えたことを特徴とするプログラマブルLSI。

【請求項2】 請求項1に記載のプログラマブルLSIにおいて、前記接続手段はクロスバスイッチであることを特徴とするプログラマブルLSI。

【請求項3】 請求項1に記載のプログラマブルLSIにおいて、前記複数の演算回路の中には数値演算を行う回路を含むことを特徴とするプログラマブルLSI。

【請求項4】 請求項1に記載のプログラマブルLSIにおいて、前記複数の演算回路の中には論理演算を行う回路を含むことを特徴とするプログラマブルLSI。

【請求項5】 請求項1に記載のプログラマブルLSIにおいて、前記演算ユニットは外部から与えられる演算の内容を指示する命令を記憶しておくメモリを有し、該メモリに記憶された命令の指示する演算の内容に対応して、前記複数の演算回路を選択することを特徴とするプログラマブルLSI。

【請求項6】 請求項5に記載のプログラマブルLSIにおいて、前記メモリは複数の前記命令を記憶するための複数の記憶領域を有し、当該複数の記憶領域から順次に前記命令を読み出し、当該読み出した命令に応じて選択された演算回路により演算を行うことを特徴とするプログラマブルLSI。

【請求項7】 請求項1に記載のプログラマブルLSIにおいて、前記接続手段は前記複数の演算ユニットをマトリクス形態で接続可能とすることを特徴とするプログラマブルLSI。

【請求項8】 請求項1に記載のプログラマブルLSIにおいて、前記複数の演算ユニットをツリー構造で接続可能とすることを特徴とするプログラマブルLSI。

【請求項9】 請求項8に記載のプログラマブルLSIにおいて、前記接続手段は前記ツリー構造の階層数を可変設定することにより前記複数の演算ユニットを選択的に接続することを特徴とするプログラマブルLSI。

【請求項10】 請求項9に記載のプログラマブルLSIにおいて、前記接続手段は前記ツリー構造の各階層上に位置し、上または下の階層で隣接する特定の演算ユニットからの演算結果を選択する選択回路を有し、該選択回路により演算結果を選択することにより前記ツリー構造の階層数を可変設定することを特徴とするプログラマブルLSI。

【請求項11】 種類の異なる演算回路複数をそれぞれが有し、演算の実行に供する演算回路の指定を外部から受け付け、演算に使用するデータを入力し、演算結果を

出力する複数の演算ユニットと、前記複数の演算ユニットを予め定めた演算式に従って相互に接続するための接続回路とをLSI化し、前記複数の演算ユニットで実行させる演算回路および前記クロスバスイッチにより接続する演算ユニットを予め定めた演算式に従って指示することを特徴とするプログラマブルLSIの演算方法。

【請求項12】 請求項11に記載のプログラマブルLSIの演算方法において各前記演算ユニットに与える演算回路の種類を示す初期データを順次に変更することにより前記演算式の内容を順次に変更することを特徴とするプログラマブルLSIの演算方法。

【請求項13】 請求項11に記載のプログラマブルLSIの演算方法において、遺伝的アルゴリズムの手法を使用して好適な演算式の内容を決定することを特徴とするプログラマブルLSIの演算方法。

【請求項14】 請求項11に記載のプログラマブルLSIの演算方法において、前記演算ユニットのそれぞれは演算中および演算終了を示すフラグ情報を演算状態に応じてセットし、該演算ユニットに接続する下流側の演算ユニットは前記フラグに基づき接続の上流側の演算結果を取り込むことを特徴とするプログラマブルLSIの演算方法。

【請求項15】 請求項11に記載のプログラマブルLSIの演算方法において、前記複数の演算回路ユニットの各々は演算の内容を指示する命令を記憶しておくメモリを有し、該メモリに記憶された命令の指示する演算の内容に対応して、前記複数の演算回路を選択することを特徴とするプログラマブルLSIの演算方法。

【請求項16】 請求項11に記載のプログラマブルLSIの演算方法において、前記メモリは複数の前記命令を記憶するための複数の記憶領域を有し、当該複数の記憶領域から順次に前記命令を読み出し、選択された演算器により演算を行うことを特徴とするプログラマブルLSIの演算方法。

【請求項17】 請求項11に記載のプログラマブルLSIの演算方法において、前記接続回路は前記複数の演算ユニットをマトリクス形態で接続可能とすることを特徴とするプログラマブルLSIの演算方法。

【請求項18】 請求項11に記載のプログラマブルLSIの演算方法において、前記複数の演算ユニットをツリー構造で接続可能とすることを特徴とするプログラマブルLSIの演算方法。

【請求項19】 請求項18に記載のプログラマブルLSIの演算方法において、前記接続回路は前記ツリー構造の階層数を可変設定することにより前記複数の演算回路を選択的に接続することを特徴とするプログラマブルLSIの演算方法。

【請求項20】 請求項19に記載のプログラマブルLSIにおいて、前記接続手段は前記ツリー構造の各階層上に位置し、上または下の階層で隣接する特定の演算回

路からの演算結果を選択する選択回路を有し、該選択回路により演算結果を選択することにより前記ツリー構造の階層数を可変設定することを特徴とするプログラマブルLSIの演算方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は演算式をプログラマブル（プログラム可能）なプログラマブルLSI（大規模集積回路）およびその演算方法に関する。

【0002】

【従来の技術】従来この種LSIとしてはFPGA（Field Programmable Gate Array）がよく知られている。FPGAはプログラム可能な複数の論理ユニットとこれらユニットを選択的に接続するクロスバースイッチから構成されている。論理ユニットはアンド（AND）、オア（OR）等の論理演算を行う。

【0003】なお、四則演算などを行う回路としてはCPU（Central Processing Unit）、DSP（Digital Signal Processor）もしくは演算式に従って加減算器を組み合わせて専用的に固定接続したLSIが知られている。

【0004】

【発明が解決しようとする課題】上記従来例の中でFPGAは論理演算がよく行なわれ、数値演算は難しい。一方、CPUやDSPは数値演算は可能であるものの、数値演算を規定した演算式はプログラムで規定されるので、演算式を変更するためにはプログラム全体を書き換えるという手間が必要となる。また、加減算等の演算器を組み合わせたLSIは演算式の変更ができない、すなわち、プログラマブルではないという欠点を有する。

【0005】そこで、本発明の目的は、上述の欠点に鑑みて、数値演算式をプログラムするのに好適なプログラマブルLSIおよびその演算方法を提供することにある。

【0006】

【課題を解決するための手段】このような目的を達成するために、請求項1の発明は種類の異なる演算回路複数のそれぞれが有し、演算の実行に供する演算回路の指定を外部から受け付け、演算に使用するデータを入力し、演算結果を出力する複数の演算ユニットと、前記複数の演算ユニットを予め定めた演算式に従って相互に接続するための接続手段とを具えたことを特徴とする。

【0007】請求項2の発明は、請求項1に記載のプログラマブルLSIにおいて、前記接続手段はクロスバースイッチであることを特徴とする。

【0008】請求項3の発明は、請求項1に記載のプログラマブルLSIにおいて、前記複数の演算回路の中には数値演算を行う回路を含むことを特徴とする。

【0009】請求項4の発明は、請求項1に記載のプロ

グラマブルLSIにおいて、前記複数の演算回路の中には論理演算を行う回路を含むことを特徴とする。

【0010】請求項5の発明は、請求項1に記載のプログラマブルLSIにおいて、前記演算ユニットは外部から与えられる演算の内容を指示する命令を記憶しておくメモリを有し、該メモリに記憶された命令の指示する演算の内容に対応して、前記複数の演算回路を選択することを特徴とする。

【0011】請求項6の発明は、請求項5に記載のプログラマブルLSIにおいて、前記メモリは複数の前記命令を記憶するための複数の記憶領域を有し、当該複数の記憶領域から順次に前記命令を読み出し、当該読み出した命令に応じて選択された演算回路により演算を行うことを特徴とする。

【0012】請求項7の発明は、請求項1に記載のプログラマブルLSIにおいて、前記接続手段は前記複数の演算ユニットをマトリクス形態で接続可能とすることを特徴とする。

【0013】請求項8の発明は、請求項1に記載のプログラマブルLSIにおいて、前記複数の演算ユニットをツリー構造で接続可能とすることを特徴とする。

【0014】請求項9の発明は、請求項8に記載のプログラマブルLSIにおいて、前記接続手段は前記ツリー構造の階層数を可変設定することにより前記複数の演算ユニットを選択的に接続することを特徴とする。

【0015】請求項10の発明は、請求項9に記載のプログラマブルLSIにおいて、前記接続手段は前記ツリー構造の各階層上に位置し、上または下の階層で隣接する特定の演算ユニットからの演算結果を選択する選択回路を有し、該選択回路により演算結果を選択することにより前記ツリー構造の階層数を可変設定することを特徴とする。

【0016】請求項11の発明は、種類の異なる演算回路複数のそれぞれが有し、演算の実行に供する演算回路の指定を外部から受け付け、演算に使用するデータを入力し、演算結果を出力する複数の演算ユニットと、前記複数の演算ユニットを予め定めた演算式に従って相互に接続するための接続回路とをLSI化し、前記複数の演算ユニットで実行させる演算回路および前記クロスバースイッチにより接続する演算ユニットを予め定めた演算式に従って指示することを特徴とする。

【0017】請求項12の発明は、請求項11に記載のプログラマブルLSIの演算方法において、最上流の前記演算ユニットに与える初期データを順次に変更することにより予め定めた演算式の解を取得することを特徴とする。

【0018】請求項13の発明は、前記解を取得するために遺伝的アルゴリズムの手法を使用することを特徴とする。

【0019】請求項14の発明は、請求項11に記載の

プログラマブルLSIの演算方法において、前記演算ユニットのそれぞれは演算中および演算終了を示すフラグ情報を演算状態に応じてセットし、該演算ユニットに接続する下流側の演算ユニットは前記フラグに基づき接続の上流側の演算結果を読み取ることを特徴とする。

【0020】請求項15の発明は、請求項11に記載のプログラマブルLSIの演算方法において、前記複数の演算回ユニットの各々は演算の内容を指示する命令を記憶しておくメモリを有し、該メモリに記憶された命令の指示する演算の内容に対応して、前記複数の演算回路を選択することを特徴とする。

【0021】請求項16の発明は、請求項11に記載のプログラマブルLSIの演算方法において、前記メモリは複数の前記命令を記憶するための複数の記憶領域を有し、当該複数の記憶領域から順次に前記命令を読み出し、選択された演算器により演算を行うことを特徴とする。

【0022】請求項17の発明は、請求項11に記載のプログラマブルLSIの演算方法において、前記接続回路は前記複数の演算ユニットをマトリクス形態で接続可能とすることを特徴とする。

【0023】請求項18の発明は、請求項11に記載のプログラマブルLSIの演算方法において、前記複数の演算ユニットをツリー構造で接続可能とすることを特徴とする。

【0024】請求項19の発明は、請求項18に記載のプログラマブルLSIの演算方法において、前記接続回路は前記ツリー構造の階層数を可変設定することにより前記複数の演算回路を選択的に接続することを特徴とする。

【0025】請求項20の発明は、請求項19に記載のプログラマブルLSIの演算方法において、前記接続手段は前記ツリー構造の各階層上に位置し、上または下の階層で隣接する特定の演算回路からの演算結果を選択する選択回路を有し、該選択回路により演算結果を選択することにより前記ツリー構造の階層数を可変設定することを特徴とする。

【0026】

【発明の実施の形態】以下、図面を参照して本発明の実施の形態を詳細に説明する。図1は本実施の形態のLS

IF (cos(X+Y) > sin(X*Z')) then
Z' ← Y else (X+Y) / Y

を示す。この式はcos(X+Y)の値がsin(X*Z')の値よりも大きいときにはZ' ← Yの値を演算結果とし、そうでない場合には(X+Y) / Yの値を演算結果とすることを意味する。

【0031】このような演算式を従来の演算器の組み合わせで示した例を図2に示す。従来では、図2に示す回路をLSI化すると、演算式をを変更することは容易ではないが本実施の形態では、自由に演算式を組み替える

I化する回路の基本構成を示す。図1において1~15は関数ユニット(PFU)である。各関数ユニット(本発明の演算ユニット)は同一のものを使用できる。関数ユニットの各々は演算内容が異なる複数の演算回路を有している。四則演算回路等などは、メモリで構成することができる。メモリ内には加算用、減算用、乗算用、減算用、SINの演算用、COSの演算用等約10種類の関数テーブルが格納されている。

【0027】加算用の関数テーブルはたとえば5+1の場合、数値5、1により定まるメモリ領域に加算結果の数値6が格納されている。したがって、使用したい関数テーブルを指定するデータと、演算対象のデータ(上記加算の場合、数値5、1)をメモリのアドレス線に入力すると、そのアドレスで指定されたメモリ領域から演算の結果がメモリのデータ線に出力される。上記各PFUは上述の演算テーブルの他、定数テーブルをも格納しており、定数のみの出力が可能である。また、AND、OR等の論理演算テーブルやIF THEN形式の論理演算を実行する演算回路をも有している。これらのメモリ、演算器は外部から指示されるデジタル信号によりマルチプレクサ等の切替器により実行に供する演算回路が選択される。なお、本実施の形態ではメモリを使用する演算回路を例としたがその他の形態の演算回路を使用してもよいこと勿論である。

【0028】各PFUはクロスバースイッチ(図中○印で図示)により選択的に相互接続される。なお●で示した箇所は接続を表し、たとえば、第1列目(1st Column)に位置するPFU2は入力データxとyに接続され、これらのデータを入力することを示している。第2列目(2nd Column)のPFU6は第1列目のPFU2と接続し、PFU2の演算出力を入力することを示している。

【0029】したがって、ユーザは各PFUに実行させる演算の種類を不図示のディップスイッチや他の指示装置を用いて指定し、クロスバースイッチを操作することで任意の演算式を設定することができる。ちなみに図1の接続例で設定された演算式は

【0030】

【数1】

ことができる点に注意されたい。

【0032】このようにプログラムされた演算式を実行する時間は次のとおりとなる。sin計算に5ユニットタイム、cos計算に5ユニットタイム、減算に1ユニットタイム、if then計算に2ユニットタイム、乗算に2ユニットタイム、除算に3ユニットタイム要すると、各PFUで行われる演算のタイミングと、全体の演算に要する時間は図3に示すようになる。図3

に示すように本実施の形態では演算式を構成する部分演算を並列的に実行できるので、CPUのようにシリアル的に部分演算を実行していく演算装置よりも本実施の形態の方が処理速度が速い。

【0033】次にプログラマブルLSIの具体的な構成例を図9を使用して説明する。

【0034】図9においてPFUBロック100は15組みのPFUを有するチップである。PFUコントローラ200およびシステムコントローラ300がPFUBロック100と対で用意される。システムコントローラ300は外部からチップセレクト信号（CS信号と略記する）、リード信号（RD信号と略記する）またはライト信号（WR信号と略記する）およびシステムアドレス信号を入力し、CS信号により選択されたPFUBロック100に対してRD信号またはWR信号を供給する。RD信号はPFUBロック内のPFUインストラクションメモリやPFU内の後述のレジスタのアドレスを指定する信号であり、RD信号の発生時にSystem addr信号によりアドレスが指定されたインストラクションメモリ（各PFU内に設置）の記憶領域（複数）や上記レジスタから、データが読み出され、System Data信号線に出力される。読み出されるデータは通常は、他のチップ（PFUBロック）に引き渡すデータである。

【0035】逆にWD信号の発生時にはSystem Addr信号によりアドレスが指定されたインストラクションメモリの記憶領域や上記レジスタに対して、実行プログラム命令や演算データ（オペランド）が書き込まれる。インストラクションメモリ116は各PFUについての実行プログラム命令をおよびデータを記憶する。実行プログラム命令には、演算命令、制御命令、即値命令の3種の命令がある。演算命令は本実施の形態では加減乗除など8種の演算を行うための関数が定義されている。

【0036】制御命令は特定のレジスタに対する演算データや演算結果の入出力の指示や、小数点位置の指定、分岐条件の指定等を行う。即値命令は演算すべきデータである。

【0037】IN0～IN7はPFUBロック100で処理すべき8組の入力データである。

【0038】PFUコントローラ100はSystem data信号により転送される情報を解析して後述の信号を作成し、PFUBロック100内のPFUに対して供給する。この供給信号について説明する。

【0039】func信号は各PFUが演算実行可能な8つの関数のうちのどの関数を選択するかを示す信号で、各PFUに対してfunc信号が送られる。たとえば、図16（e）に示すような3つのPFUが接続されている状態で、左側のPFUに対するfunc信号の内容を書き換えることによりそのPFUは加算から乗算に

演算実行する関数を変更する。変更結果が図16（f）になる。

【0040】PFU go信号は全PFUに対して動作開始を指示する信号である。

【0041】PFU sel信号は動作を許可する信号で各PFUに対して供給され、この信号がセットされていないPFUは動作することができない。

【0042】mux信号はPFUBロック100内の複数のPFUの接続構成を指示する信号である。この信号により複数のPFUの信号線の接続／否接続が制御されて、たとえば、図15の（a）、（b）、（c）、（d）のような各種のツリーの接続構造が構築される。

【0043】PFU done信号は各PFUから送られる実行終了信号である。

【0044】CLK enable信号はCLK（基準クロック信号）から作成された動作タイミング信号であり、この信号に同期してPFUが動作する。

【0045】PFUBロック100内のPFUに関する主要構成を図10に示す。図10において、PFU0、PFU1のそれぞれの演算結果をPFU8に入力し、PFU2、PFU3のそれぞれの演算結果をPFU9に入力するというようにして、図10の形態では4層の階層（ツリー）でPFUが接続されている。さらに図16において左端の各階層の隣接の特定のPFU（PFU0、PFU8、PFU12、PFU14）の演算結果を取り出すことができる。この取り出した演算結果中からマルチプレクサ（MPXと略記、セクタとも呼ばれる）101により所望のものを取り出す。この実施の形態では2つのPFUの間の信号線を接続／断を行うスイッチがない点に注目されたい。

【0046】どのPFUの演算結果をMPX101が取り出すかは上述のmux信号により指示される。たとえば、図15の（a）のようなツリー構造で演算を行いたい場合には、MPX101においてPFU14の演算結果を選択して出力すればよい。図15の（b）のようなツリー構造で演算を行いたい場合にはPFU12の演算結果を選択する。図15（c）のようなツリー構造の場合にはPFU8の演算結果を選択し、図15（d）のツリー構造の場合にはPFU0の演算結果を選択する。

【0047】このような接続構成とすることで、スイッチ群が不要となり、回路構成が簡素化される。また、複数のPFUで所定の演算式を組む場合にも各PFUとそのPFUに割り当てる関数との対応関係を把握することが容易という利点がある。

【0048】上述のPFUの構成の一例を図11に示し、内部構成を示す。

【0049】PFUは主にデータ入力用のFIFOバッファ110、レジスタ群111、乗算器（MPU）114、論理演算ユニット（ALU）115およびPFUインストラクションメモリ116から構成される。上述の

System addr 信号によるアドレス指定により System data 信号の内容が外部から書き込まれる。プログラムカウンタ用のレジスタ PC を介してアドレス制御回路 (Next Address Control) からのアドレス指定により、インストラクションメモリ 116 に格納された命令が順次に読み出され、デコーダ 117 により命令の内容が解析される。この解析結果により、MPU 114、ALU 115 およびレジスタ群 111 が制御され、選択された関数による演算が行われる。

【0050】アドレススタックはサブルーチン (分岐命令) が与えられたときに戻りアドレスを記憶しておくための記憶回路であり、スタックポインタ (SP) の指示するアドレスの戻りアドレスがアドレススタックから読み出される。上述したように PFU は 8 種の関数の演算が可能なので、選択された関数の種類により MPY 114、ALU 115 または MPY 114 および ALU 115 の双方が選択される。

【0051】この演算に関連して、外部入力用の FIFO 110 およびレジスタ群 111 の中の所定のレジスタから演算に使用するデータがバス A-BUS、B-BUS、C-BUS を介して MPY 114 および ALU 115 に転送される。また、MPY 114 の演算結果を ALU 115 に入力することも可能である。上述のレジスタや FIFO からのデータ入力のためにセクタ群 112、113 が使用される。

【0052】この例では MPY 114 は入力 R、S を持ち、演算結果はレジスタ M に格納され、Mout → MO ~ M2 の経路でレジスタ MO ~ M2 に MPY 114 の演算結果を格納することが可能である。ALU 115 は入力 U、V を持ち、演算結果はレジスタ A に格納され、レジスタ Aout → AO ~ A2 の経路でレジスタ AO ~ A2 に ALU 115 の演算結果を格納することが可能である。これらの演算結果を出力する時にシフトが小数点位置の調整に使用される。レジスタ M、レジスタ A および MPY 114、ALU 115 は共に外部リセット信号あるいはアドレス制御回路からの信号によりレジスタ Mrs、Ars を介してリセットされる。

【0053】レジスタ X、Y は外部入力のデータを格納し、レジスタ c0 ~ c8 は定数を格納する。本実施の形態では前の実施形態で説明した染色体データを格納する。レジスタ iml には即値を格納する。

【0054】これらの構成部は CLK enable 信号から作成されたイネーブル信号 (EN の表記を有する信号、たとえば、ENO、ENX 等) により動作可能となる。

【0055】上述の回路の入出力関連の動作説明を図 12、図 13、図 14 を参照して説明する。図 12 は外部から情報入力するときの信号発生タイミングを示す。図 13 は演算結果を出力する場合の信号発生タイミングを

示す。図 14 は PFU の処理開始と停止を行う場合の信号発生タイミングを示す。

【0056】図 12 において、時刻 T1 でのインストラクションメモリ 116 から読み出された命令が入力命令のとき (レジスタ IR の格納命令が入力)、ENX 信号によりレジスタ X 側の FIFO に情報が入力され、時刻 T3 で ENY 信号によりレジスタ Y 側の FIFO に情報が入力される。時刻 T2 での Empty X 信号の発生、時刻 T4 での Empty Y 信号および ENI 信号の発生に応じて 2 つの FIFO 110 からレジスタ X、Y に入力情報に転送される。

【0057】一方、図 13 に示すようにインストラクションメモリ 116 から出力命令が読み出されると (レジスタ IR の格納命令が出力)、CLK enable 信号に同期して ENO 信号が発生し、レジスタ Mout またはレジスタ Aout から演算結果が出力される。

【0058】動作開始にあつたつては図 14 に示すように PFU go 信号の発生に応じてプログラムカウンタ用レジスタ PC の値が順次にインクリメントされて、インストラクションメモリ 116 から順次にプログラム命令が読み出される。PFU go 信号の消去に応じて FIFO reset 信号が発生され、FIFO 110 がリセットされる。また、他の回路も動作を停止する。

【0059】演算に関連する動作タイミングは従来の演算回路と同様であるので、詳細な説明を要しないであろう。

【0060】以上、説明したように上述の実施形態では複数の PFU をツリー構造で接続し、その階層を可変設定可能とすることによりマトリクス形態の PFU の接続構造に比べて、接続構成をより簡素化できる。

【0061】なお、図 10 に示した例では PFU を 15 組み有する PFU ブロックの例を示したがこの例に限定することなく、PFU は所望の組み数とすることができる。上述の実施の形態の他に次の例を実現できる。

【0062】1) 本実施の形態では演算式をプログラムし、演算式に代入する入力データを与えることで演算式の演算結果を取得する例であるが、入力データと演算結果のデータの組を繰り返し与えることで、演算式を学習により変更させることができる。

【0063】このようにして好適な演算式を求める際に遺伝的アルゴリズム (Genetic Algorithm) の手法を使用することができる。このアルゴリズムは本願発明者によりすでに提案されて発表されているが (「進化するハードウェア」 1995 年 BIT (10 月号)) この提案内容を簡単に紹介しておく。

【0064】各演算ユニット内の選択された演算回路の種類と演算ユニット間の接続関係を解とし、この解の候補を、染色体 (0、1 の 2 進ビット列) として表現し、これを複数個用意して初期集団とする。また、これらの解の候補を個体と呼ぶ。たとえば、図 4 に示す曲線上の

黒丸の一つを図5に見られるような0と1の2進ビット列として表現し、これを複数個(図5では4個)用意する。

【0065】解の良さを定義する評価関数を定義し、その値を適応度とする。解く問題ごとに最適な評価関数を用意する。

【0066】適応度の高い個体同士を集団の中から選択する。その選択の方法として一般的なものは比例配分による方法で集団中の適応度の総和において、各個体の占める割合に応じて各個体が選ばれる。図5、図6に割合を示す。

【0067】交差(データの一部が図7に示すように交互に移動したもの)や突然変異(データの一部が他のものに変化したもの)を上記選択した2つの個体に適用してさらによい個体を作り出す。このようにして作り出された解の中で評価関数により定まる適用度の低い解は淘汰して解の集団数を一定に保ちつつ、適用度が高くて満足する解が得られるまで上記個体の選択、個体の作り換えを繰り返す。

【0068】この結果、プログラマブルLSIにおいて設定される演算式が自動的に変更され、最終に好適な演算式が得られる。このプログラマブルLSIの用途としてはパターン認識に加えてたとえば、暗号解読や、データの統計の解析、通信におけるデータ圧縮、ATM(交換機)の制御に使用できる。

【0069】2) 通常、加算や乗算など演算の種類に応じて演算時間が異なる。そこで、図1の各PFUでは演算状態を示すフラグ情報を用意して下流側のPFUに演算状態を知らせるとよい。より具体的には、ビット1で演算終了を表し、ビット0で演算中を表すフラグを上流側のPFUは演算開始時および終了時に設定する。このフラグ情報をフリップフロップやラッチ回路に設定する。下流側のPFUでは図8に示すような処理手順でこのフラグ情報を監視し(図8のステップS1→S2のループ処理)、フラグ情報が計算終了を示したときに上流側のPFUからその演算結果を入力する(ステップS3)。また、自己より下流側のPFUのために自己のフラグ情報を0に設定する。指示された種類の演算を実行して演算結果が得られると下流側のPFUへ演算結果を引き渡すための出力用バッファに計算結果をセットして、自己のフラグ情報1にセットする(ステップS4→S5)。このような処理を各PFUが行うことにより非同期でデータを転送することが可能となる。

【0070】

【発明の効果】以上、説明したように請求項1、2、3、11の発明では、演算ユニットの中の演算回路を所望の演算式に従って種類指定し、接続手段(クロスバースイッチ)により接続していくことで、数値演算式をプログラムして行くことができる。

【0071】請求項4の発明では、演算回路の中に論理

回路を含めることで数値演算結果を用いた論理演算式をプログラムすることが可能となる。

【0072】請求項5、15の発明では、各演算ユニットはメモリを有することにより、外部から異なる命令を受け付けることができ、実行する演算内容を固定化せず、変更することができる。

【0073】請求項6、16の発明では、メモリが複数の記憶領域を有することにより、異なる演算内容を指示する命令を一括してメモリに記憶して、順次に異なる演算を行うことができるので、他種、多様の演算式を実行することができ、演算式がプログラマブルとなる。

【0074】請求項7、17の発明では、演算ユニットの接続形態をマトリクス形態とすることで、ありとあらゆる演算を実行することができる。

【0075】請求項8~10、18~20の発明では、演算ユニットの接続形態をツリー構造とすることで、マトリクス形態よりも少ない演算ユニットで演算を実行することができる。また、ツリー構造の階層数を可変することにより、異なる演算式に対応することができる。また、各階層の特定の演算回路の演算結果を選択することで、信号線の接続/断を行うスイッチを特に設ける必要はなく、LSI内部の回路構成を簡素化することができる。もって、システム全体の小型化に寄与することができる。

【0076】請求項12の発明では、演算式の各演算の種類およびその接続を規定する初期データを順次に変更することで、演算式の解すなわち、好適な演算式を取得することができる。これによりプログラマブルLSIでニューラルネットワークや、統計分析回路を構成することができ、パターン認識やデータ解析等を実行することができる。

【0077】請求項13の発明では、取得した解の中で好適な解を選択することができる。

【0078】請求項14の発明では、演算ユニット間のデータ転送を非同期で行うことができる。

【図面の簡単な説明】

【図1】本発明実施の形態の回路構成を示す構成図である。

【図2】演算式に従った演算回路の接続例を示すブロック図である。

【図3】演算処理タイミングを示す説明図である。

【図4】遺伝的アルゴリズムを説明するための説明図である。

【図5】遺伝的アルゴリズムを説明するための説明図である。

【図6】遺伝的アルゴリズムを説明するための説明図である。

【図7】遺伝的アルゴリズムを説明するための説明図である。

【図8】PFU間のデータ通信処理内容を示すフローチ

ャートである。

【図9】他の実施の形態のシステム構成を示すブロック図である。

【図10】PFUの接続構成を示すブロック図である。

【図11】PFUの内部構成を示すブロック図である。

【図12】入力の信号発生タイミングを示すタイミングチャートである。

【図13】出力の信号発生タイミングを示すタイミングチャートである。

【図14】動作開始および停止の信号発生タイミングを示すタイミングチャートである。

【図15】PFUの接続パターン例を示す説明図である。

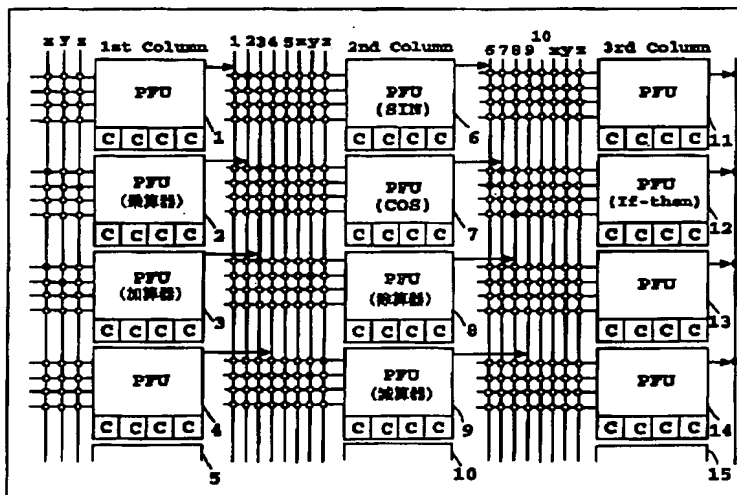
【図16】PFUの演算内容の切り換えを示す説明図で

ある。

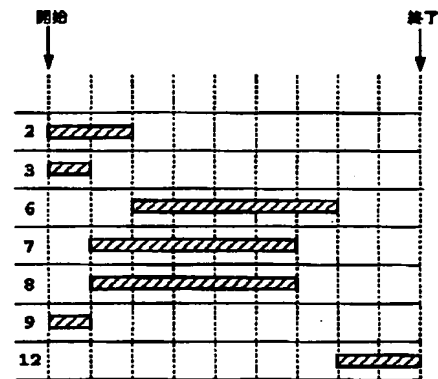
【符号の説明】

- 1～15 関数ユニット
- 100 PFUブロック
- 110 FIFO (バッファ)
- 111 レジスタ群
- 112, 113 セレクタ
- 114 乗算器 (MPU)
- 115 論理演算ユニット (ALU)
- 116 インストラクションメモリ
- 117 デコーダ
- 200 PFUコントローラ
- 300 システムコントローラ

【図1】

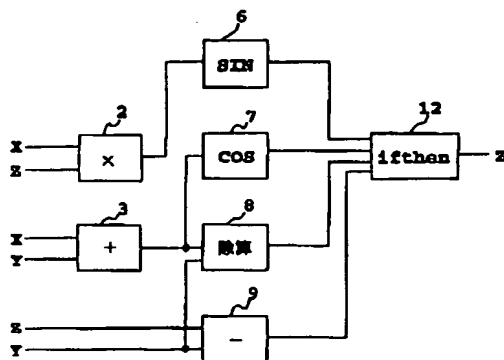


【図3】

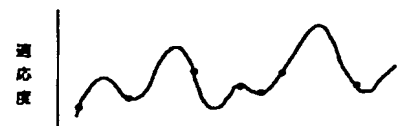
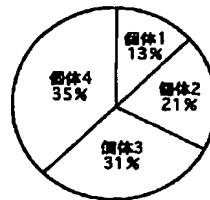


【図4】

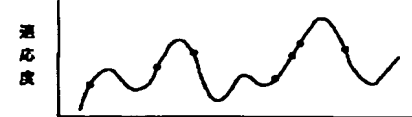
【図2】



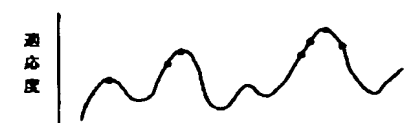
【図6】



解の分布 (時刻0)



解の分布 (時刻5)

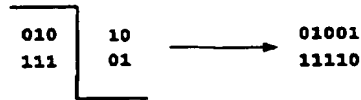


解の分布 (時刻10)

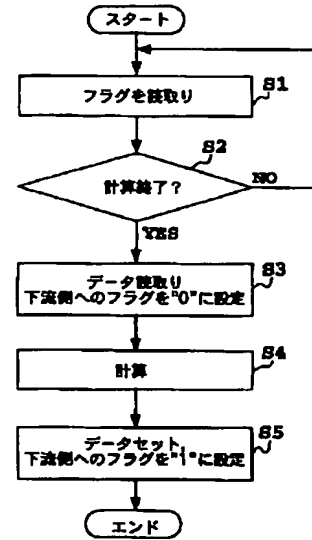
【図5】

個体	適応度	割合 (%)
00100(4)	2.0	13
01010(10)	3.2	21
10110(22)	4.7	31
11101(29)	5.4	35

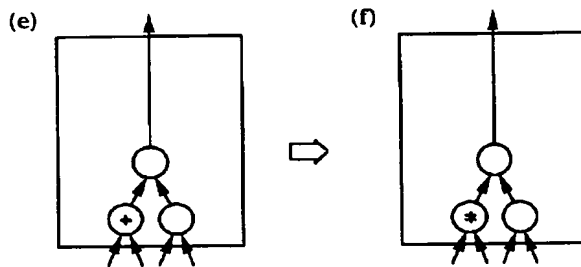
【図7】



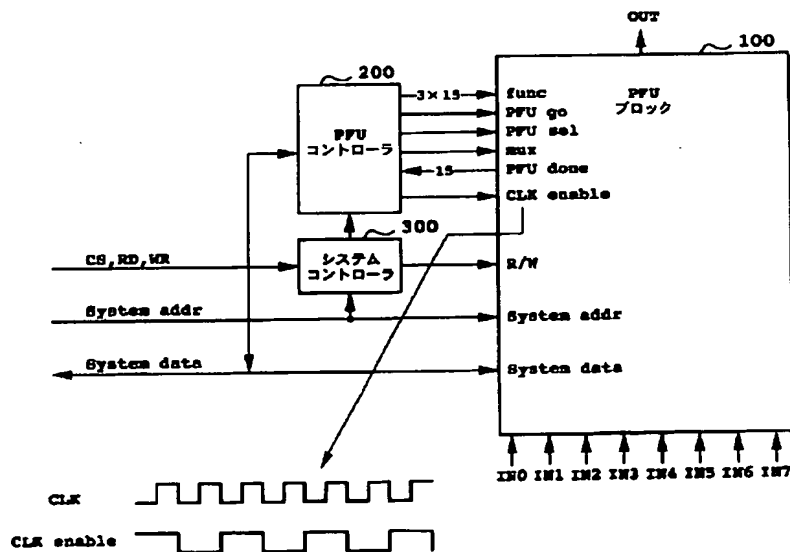
【図8】



【図16】

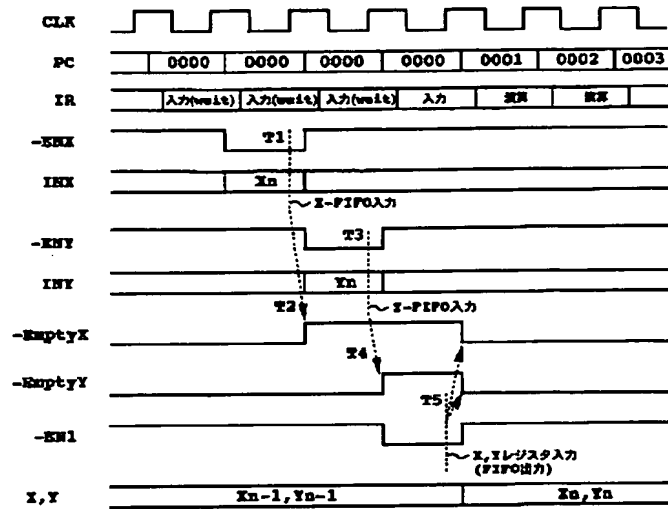


【図9】

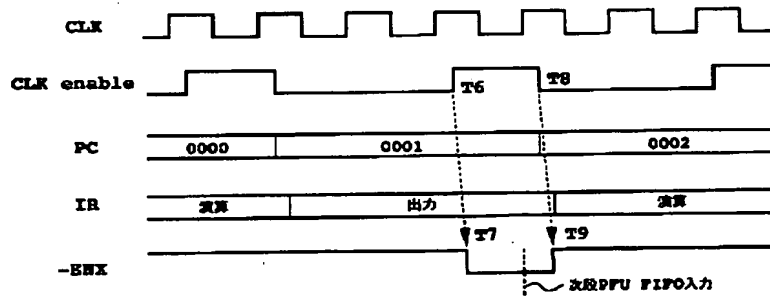


[illegible]

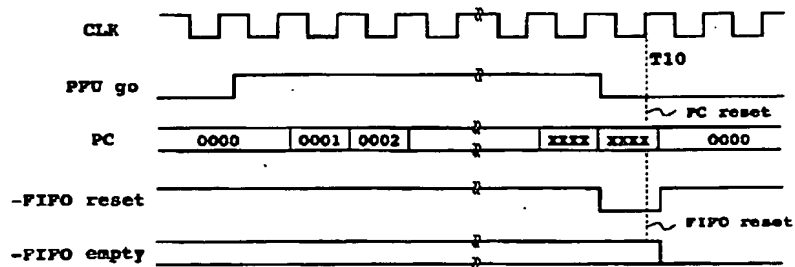
【図12】



【図13】



【図14】



【図15】

